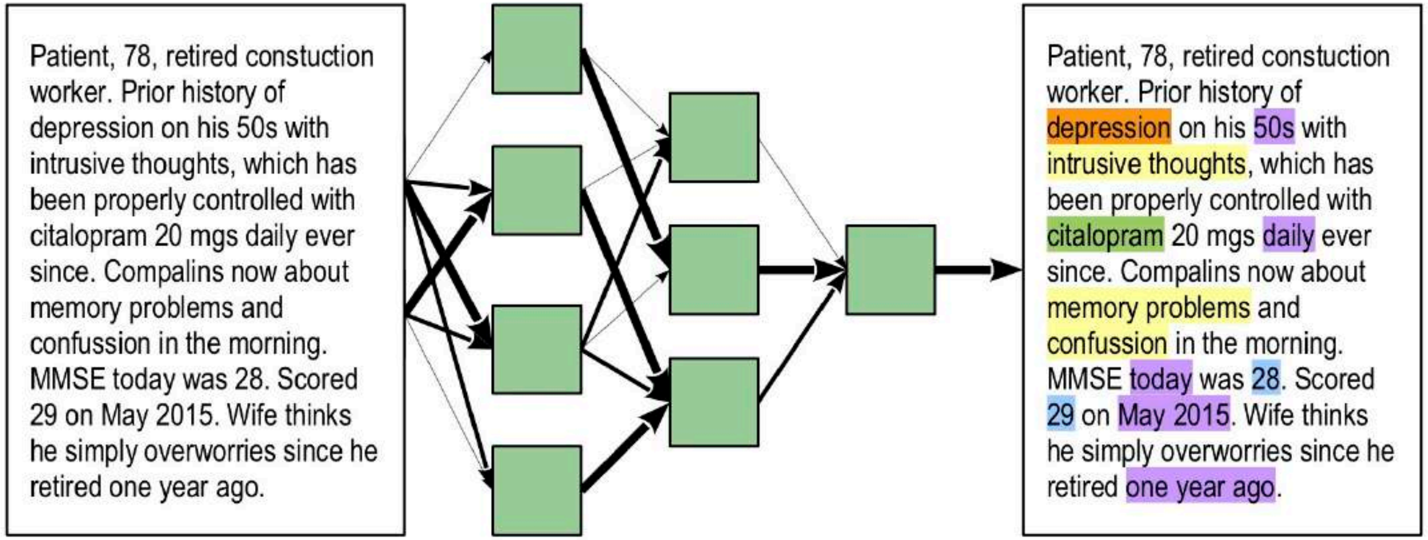
Named Entity Recognition from Medical Text with Bi-Directional LSTMs

Online version for sigmasquared.io

Maximilian Hofer

What are we trying to do? Model **sequential data**.



Text source: with permission from Oxford Translational Neuroscience & Dementia Research lab (https://www.psych.ox.ac.uk/research/dementia-research-group).

Recap: feed-forward NN vs RNN

Background

The RNN 'self-loop' gives each RNN cell a virtual memory, which is useful when processing sequential data, e.g. natural text.

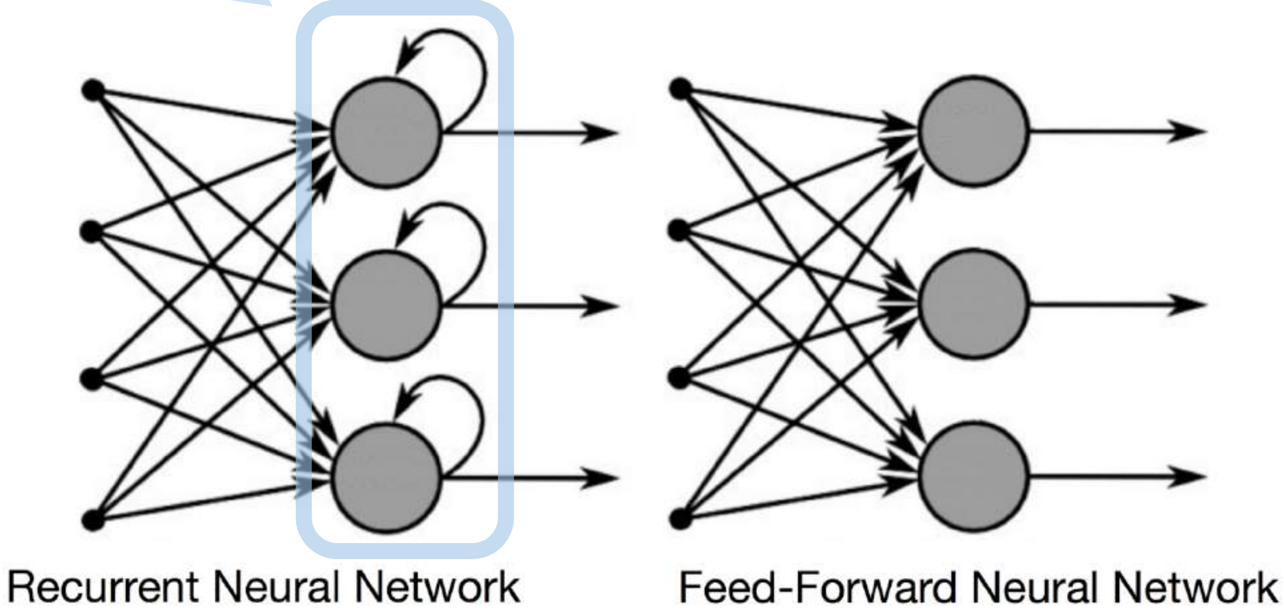


Image from *Deep Learning*. Goodfellow, Bengio and Courville (https://www.deeplearningbook.org/)

LSTM architecture: a particular type of RNN

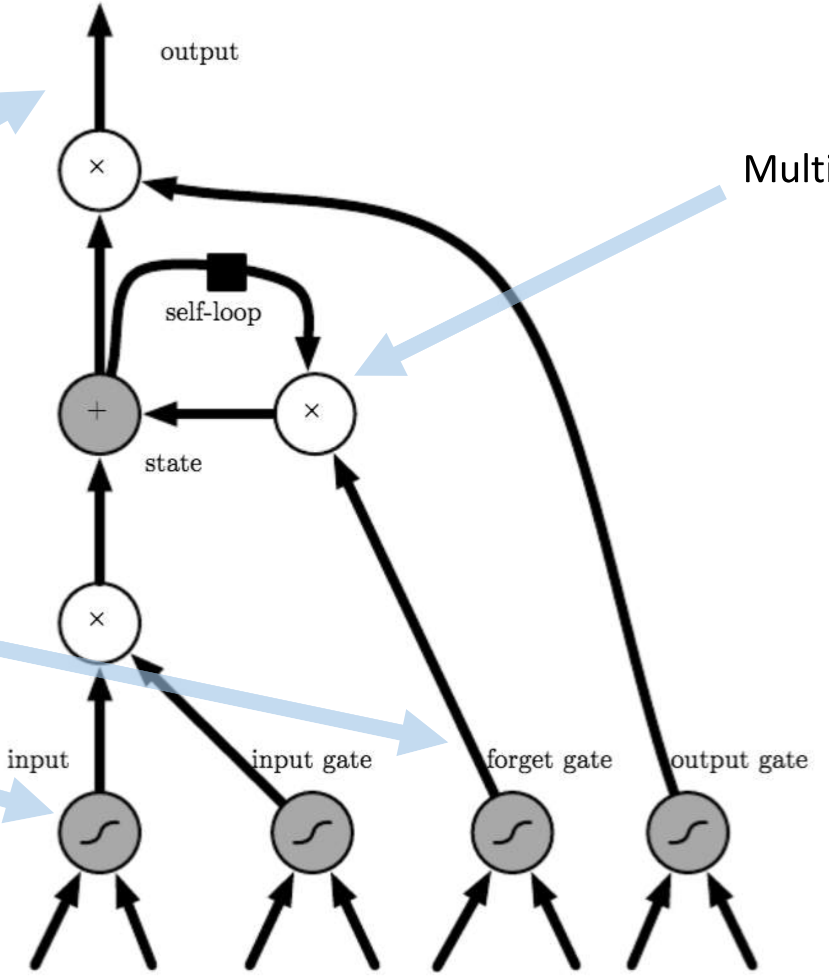
Background

Long short-term memory (LSTMs) can remember associations across long sequences of text – hence the name – *looking forward* from word at position *t*.

At each time step (i.e. word):

- Learns what to “pass on”
- Learns what to “forget”
- Sigmoid non-linearities

Multiplication



LSTM cell

Image from *Deep Learning*. Goodfellow, Bengio and Courville (https://www.deeplearningbook.org/)

BLSTM: forward LSTM → backward LSTM ←

Background

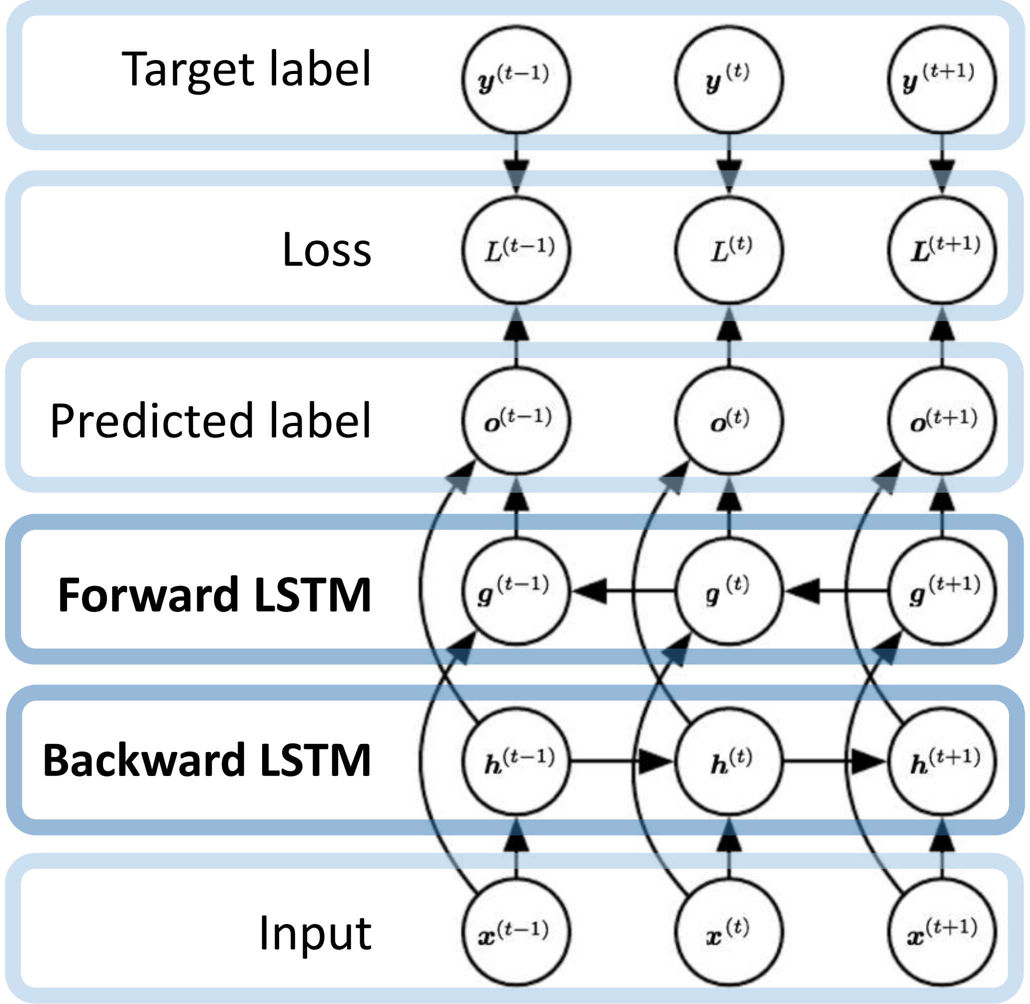
Bi-directional LSTM

capture the context

before and *after* a

given word at

position *t.*



BLSTM network

Image from *Deep Learning*. Goodfellow, Bengio and Courville (https://www.deeplearningbook.org/)

Text data: Electronic Health Records (EHRs)

Application

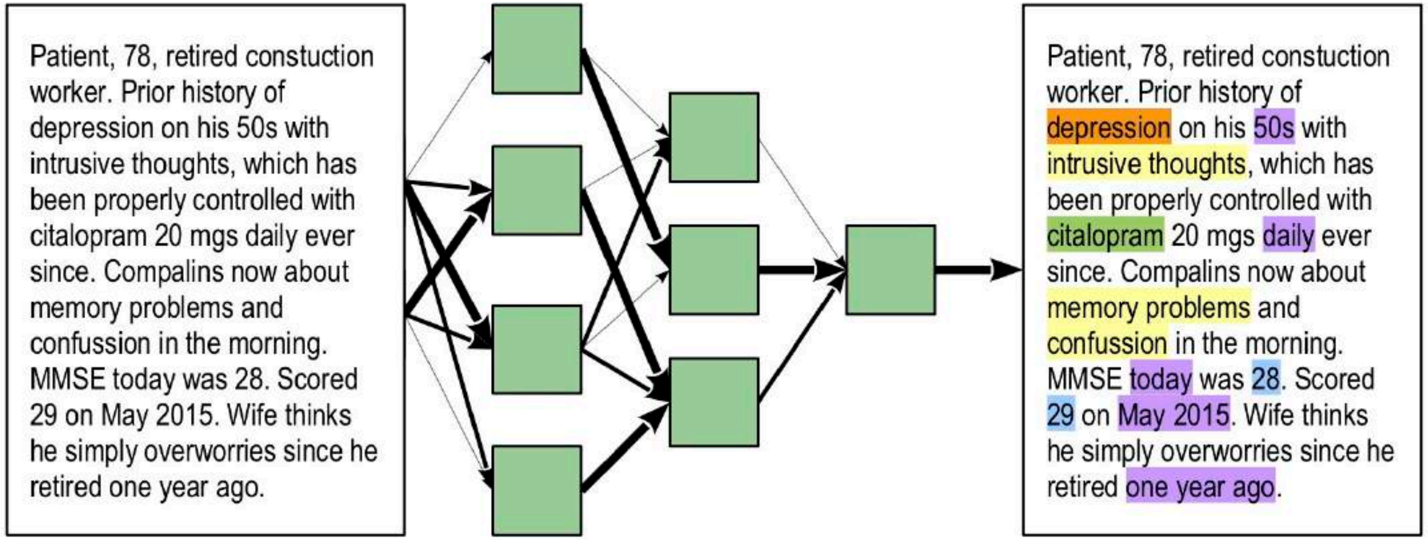
- Written by doctors and clinicians in hospitals
- Contain diagnosis, drug names, drug dosages, dates, etc.
- For example:

Patient, 89, retired construction worker. Prior history of mild depression in his 60s treated with paroxetine 20mg …

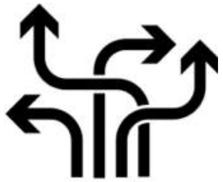
Image from https://pixabay.com/en/doctor-laptop-office-medical-male-3212067/

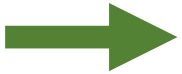
Application

Objective: identify important details



Unstructured



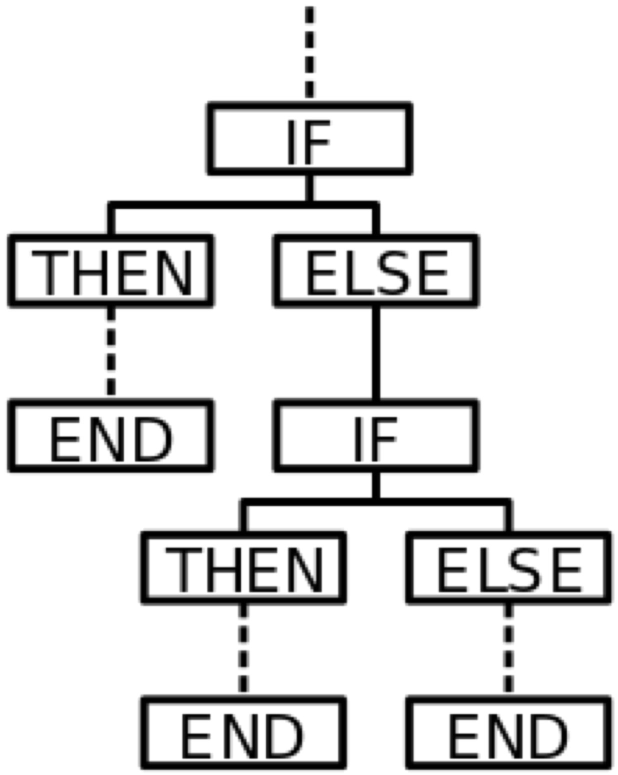


Structured

Diagnosis
Drug name
Drug dosage

Why not just use **rules**?

1. Rules are very time-consuming to code
2. Rules require expensive, expert knowledge from highly educated doctors
3. Rules are very brittle: What if a word is misspelled? Abbreviations?



Machine Learning overcomes all of these issues (given enough annotated data)!

Application

Data: discharge summaries

- i2b2 yearly NLP challenge (Harvard Medical School)
- US hospitals: *Partners Healthcare, Beth Israel Deaconess Medical Center, and the University of Pittsburgh Medical Center*
- n=170 annotated summaries
- Unbalanced target labels: more than 90% of words are not interesting
  - Target value “O” for other on the next slide

```
Although left
transmetatarsal amputation
being considered, it was
felt that she had a good
chance of healing the wound
appropriately. She had a
single temperature spike,
although all cultures
remained negative. She had
continuation of her Heparin
while she was started on a
course of Coumadin to
reserve patency of her
graft. […]
```

Example data source from Uzuner, Solti and Cadag (2010).

Application

Data preparation: word, POS, chunk and target

Word	*POS*	*Chunk*	*Target*
DISCHARGE	NNP	I-NP	**O**
MEDICATIONS:	NNP	I-NP	**O**
ACETYLSALICYLIC	NNP	I-NP	**B-m**
ACID	NNP	I-NP	**I-m**
325	CD	O	**B-do**
MG	NNP	B-NP	**I-do**

POS = point of speech tag, e.g. proper noun singular (NNP)

Chunk = larger parts of sentences, e.g. noun phrases (NP)

Target = ”B” is beginning of a sequence of words belonging to the “m” medication tag

Application

Turning words into vectors

Word vector	*POS*	*Chunk*	*Target*
-0.33 0.45 -0.07 -0.69 …	NNP	I-NP	**O**
1.47 -0.89 -0.45 -1.41 …	NNP	I-NP	**O**
NA $\sim U(-0.25, 0.25)^d$	NNP	I-NP	**B-m**
0.32 -0.23 -0.28 0.18 …	NNP	I-NP	**I-m**
0.03 -0.52 0.06 0.29 …	CD	O	**B-do**
-0.44 -1.05 -0.49 0.22 …	NNP	B-NP	**I-do**

Word vector = *d*-dimensional vector describing each word

Download e.g. Pennington, Socher and Manning (2014) **GloVE** from
https://nlp.stanford.edu/projects/glove/

Application

Input matrix for the BLSTM

Word vector	*POS*	*Chunk*	*Target*
-0.33 0.45 -0.07 -0.69 …	0	0	**0**
1.47 -0.89 -0.45 -1.41 …	0	0	**0**
0.12 -0.03 0.21 0.16 …	0	0	**1**
0.32 -0.23 -0.28 0.18 …	0	0	**2**
0.03 -0.52 0.06 0.29 …	1	1	**3**
-0.44 -1.05 -0.49 0.22 …	0	2	**4**

Number of words in training corpus

Word vector dimensions + POS + chunk + target

→ **BLSTM**

Application

Evaluating predictions: **STRICT** evaluation

Word vector	***POS***	***Chunk***	***Target***	***Pred***
DISCHARGE	0	0	**O**	**O**
MEDICATIONS:	0	0	**O**	**O**
ACETYLSALICYLIC	0	0	**B-m**	**B-m**
ACID	0	0	**I-m**	**I-m**
325	1	1	**B-do**	**B-do**
MG	0	2	**I-do**	**0**

Score of ½.

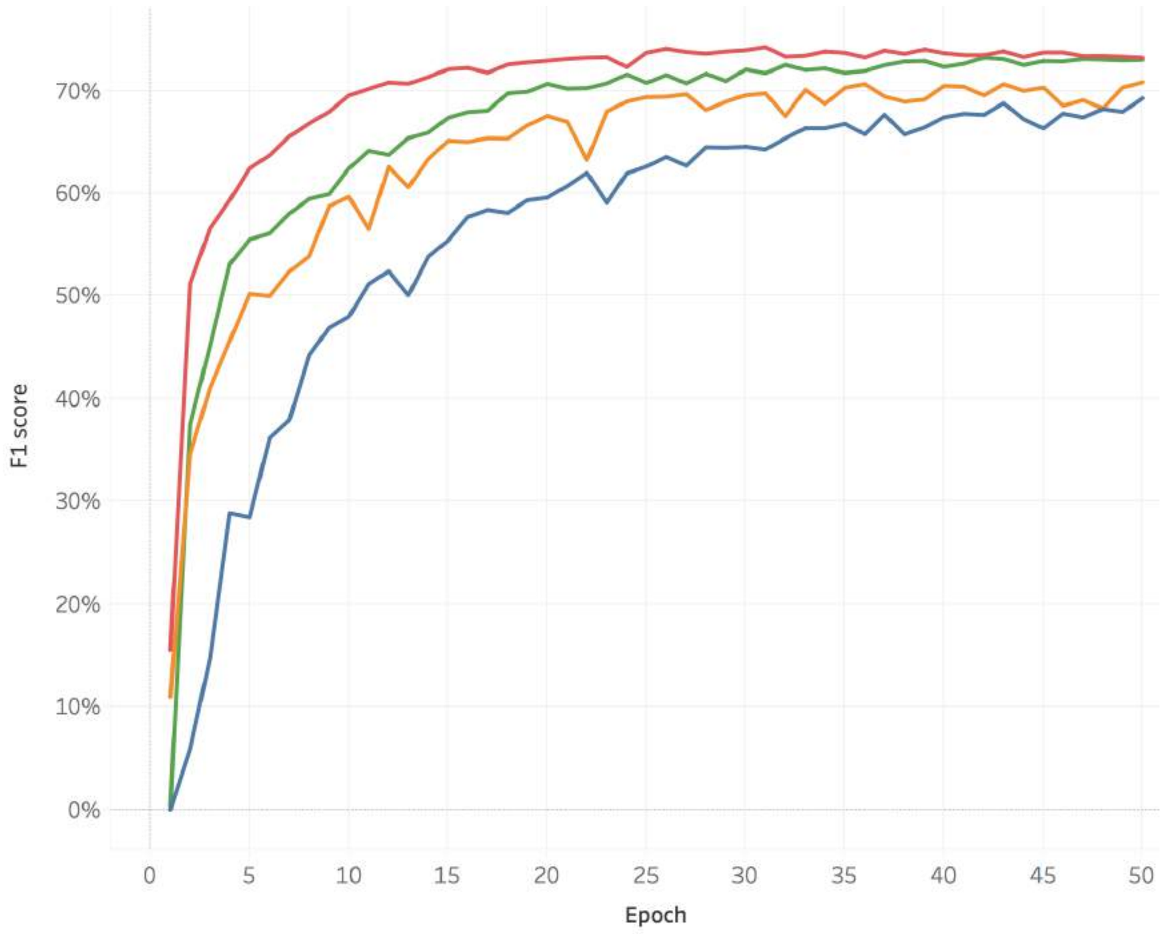
Evaluating predictions: **RELAXED** evaluation

Word vector	***POS***	***Chunk***	***Target***	***Pred***
DISCHARGE	0	0	**O**	**O**
MEDICATIONS:	0	0	**O**	**O**
ACETYLSALICYLIC	0	0	**B-m**	**B-m**
ACID	0	0	**I-m**	**I-m**
325	1	1	**B-do**	**B-do**
MG	0	2	**I-do**	**0**

Score of ¾.

For details of strict and relaxed evaluation metrics for medical NER visit Ozlem Uzuner's examples on http://faculty.washington.edu/fxia/mpapers/2009/uzuner2009_i2b2_eval.pdf

Training performance measured by F1 score



Colors represent different runs.

Details in arxiv paper:
https://arxiv.org/abs/1811.05468

Python implementation:
https://github.com/mxhofer/Named-Entity-Recognition-BidirectionalLSTM-CNN-CoNLL

Thank you for your attention.

Do you have any questions?